Bulletin MP717

A Texas Instruments Application Report

Vacuum florescent display driven by TMS9940

TABLE OF CONTENTS

VACUUM FLORESCENT DISPLAY DRIVEN BY TMS 9940

INTRODUCTION

Alphanumeric displays are required in many microprocessor-based systems. Multiposition displays provide a variety of functions, e.g., data readout, operator guidance, error messages, and the like. Vacuum fluorescent displays offer alphanumeric capability with good brightness in multicharacter packages at relatively low cost.

Figure 1 is the block diagram of a multiple position, alphanumeric display. The function of the alphanumeric display is to accept input data and produce a visible image that is a function of the information contained in the input data (e.g., recognize and display all printable ASCII characters input in groups of 20 characters at a time). The input data could be of many forms (e.g., ASCII, Hollerith, Baudot, EBCDIC), in forward or reverse order, and from different types of input sources so that, in general, some sort of communications handling and data formatting circuitry is needed to receive the incoming data, prepare it for storage, and optionally provide handshake signals for the input data source equipment.

Since even a relatively small alphanumeric display contains a large number of dots or segments to be controlled, the display will be scanned (i.e., one character at a time is illuminated) as opposed to illuminating all characters at the same time. Also it is usually a requirement that the display receive input data one time and then hold a corresponding image for some desired viewing time. This implies that the display must have a memory where image data is contained. In the diagram, this memory is called the display image buffer. The display scanner uses this information to create the characters in the display each time the display is scanned. The data residing in the display image buffer is created by the data formatter and may be written there whenever the display scanner is not reading data out of the buffer.

This report will show how the TMS9940 single chip microcomputer can make an efficient, cost-effective controller for a multicharacter fluorescent display since the functions of display refresh, communications handling, and character font assignment can be performed with one instead of several chips. In more special purpose display applications, it is conceivable that the computer could also perform other functions besides simple display control such as checking for data validity, modifying data before displaying, scrolling or flashing the display, or perhaps buffering several messages for repetitive display.

20 DIGIT ALPHANUMERIC DISPLAY

The architecture of a 20 digit alphanumeric display is shown in Figure 2. The display receives serial data (and may optionally echo it back to the sender) by way of an RS-232 connector. The data thus received is then decoded, formatted, and stored in such a manner as to make the scanning or refreshing of the display as straightforward as possible.

The main task, that of refreshing the display, is handled by the TMS9940, a single chip microcomputer. The TMS9940 further performs the tasks of data decoding, formatting, and storage (memory is provided on the chip). In conjunction with the TMS9902 Asynchronous Communications Controller, the TMS9940 also performs the tasks of receiving serial input data and optionally echoing that data back to the sender. Providing handshake signals to the sender is also optional.

The arrangement and number of connections to the display are, of course, dictated by the organization of the display. In this example, the display is made up of 20 character positions of 5 × 7 dot matrix characters. There are 20 leads for linear selection of the particular character (grid) to be illuminated

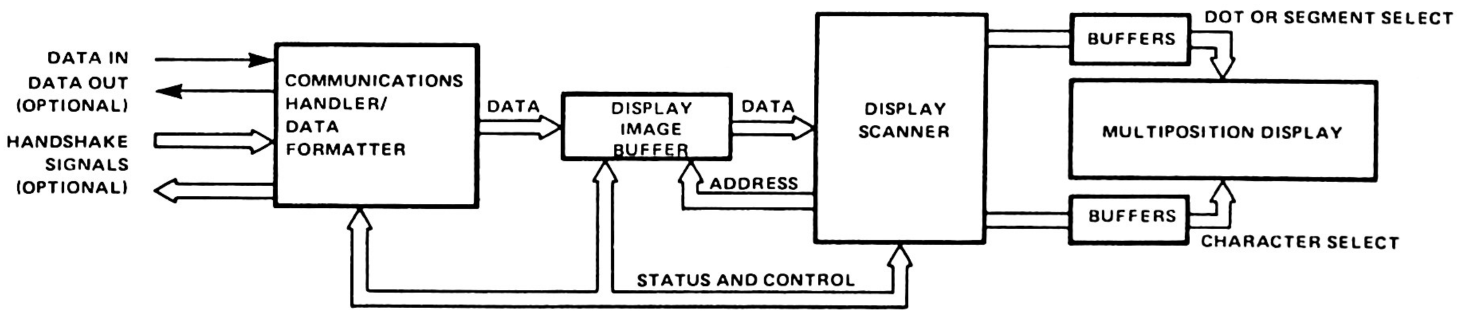

Figure 1. Alphanumeric Display Block Diagram

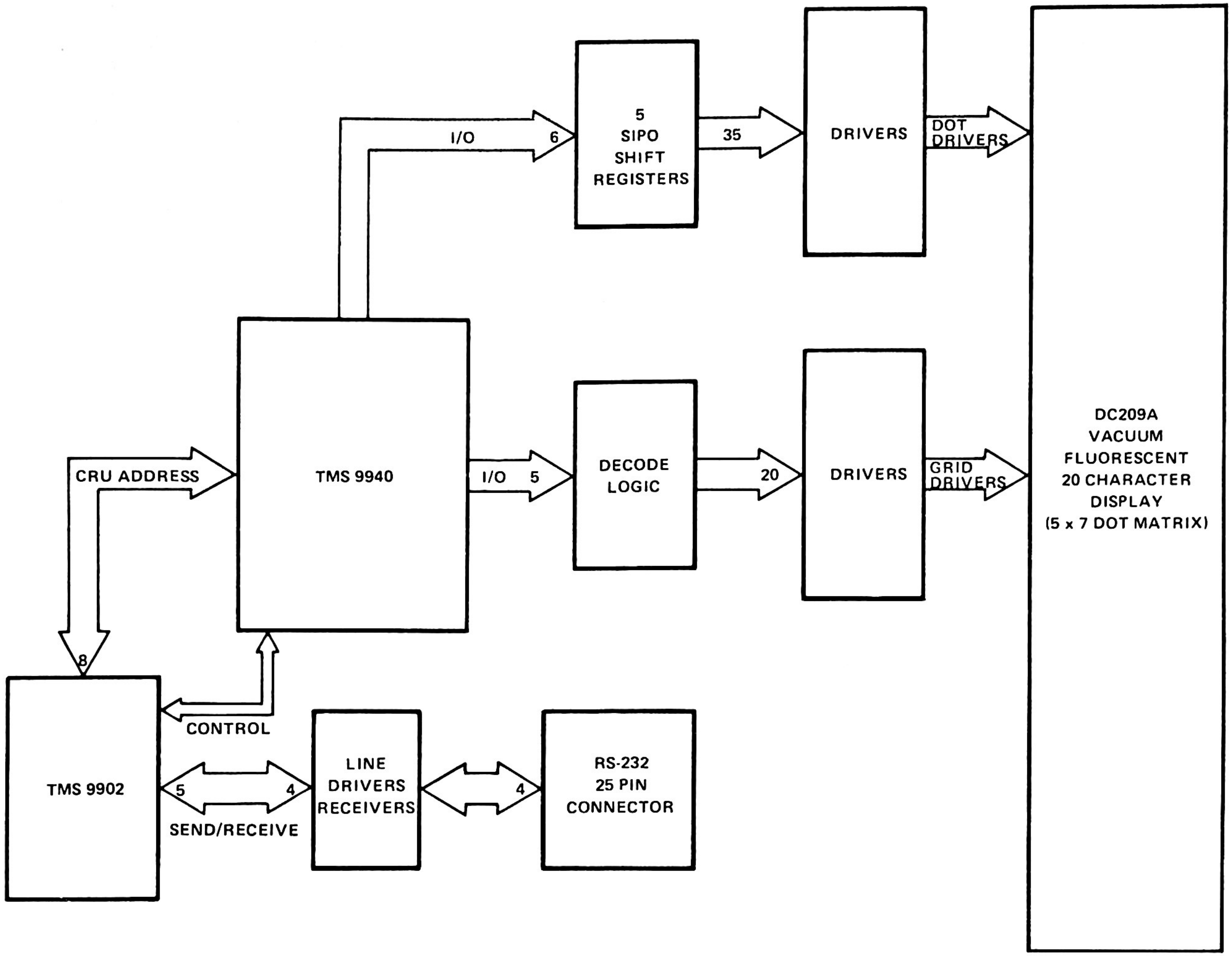

Figure 2. Vacuum Fluorescent Display Block Diagram

and 35 leads for turning on or off each dot of the active character ($7 \times 5 = 35$). The 20 grid select lines are decoded from five I/O lines of the microcomputer while the 35 dot driver lines come from five dot driver registers having seven outputs each. These five dot driver registers are loaded from six I/O lines of the microcomputer (five separate inputs plus one common clock line). The remaining connections in the system will be detailed in the following section.

The dot driver registers are serial-in, parallel-out (SIPO) register. Figure 3 shows how the outputs of these registers, when loaded from the microcomputer, are used to create the letter "T". Note that seven outputs from each SIPO register control the seven dots of one column in the character matrix and that dots are illuminated by the presence of logical 1's at the register outputs.

HARDWARE DESIGN

The TMS9940 is a complete 16-bit microcomputer on a chip, including a CPU, EPROM or ROM, RAM, clock driver, interrupts and I/O. With a 16-bit instruction set, 2048 bytes of EPROM or ROM, 128 bytes of RAM, four prioritized interrupts, on-chip Timer/Event counter, and 32 I/O ports, the TMS9940 is a very powerful one-chip computer. Figure 4 shows a simplified block diagram of the TMS9940. The instruction set is virtually identical to that of the TMS 9900.

The TMS9940 employs an advanced memory-to-memory architecture where blocks of memory designated as work-spaces replace dedicated hardware registers with program-data registers. The TMS9940 memory map is shown in Figure 5. The 2K $\times$ 8 EPROM/ROM is assigned memory addresses 0000_{16} through $07FF_{16}$, and the 128 $\times$ 8 RAM is assigned memory addresses 8300_{16} through $837F_{16}$.

The first eight words in the EPROM/ROM (addresses 0000_{16} through $000F_{16}$) are used for the interrupt vectors. Twenty-four words, addresses 0050_{16} to $007F_{16}$ are used for extended operation (XOP) instruction trap vectors. The remaining memory is available for programs, data, and workspace register. If desired, any of the special areas may also be used as general EPROM/ROM memory.

Three machine registers are accessible to the user. The 16-bit program counter (PC) and the 16-bit status register (ST) are both used in the traditional fashion. The 11-bit workspace register (WP) points to the first word in the currently active set of workspace registers. [Refer to the TMS9940 16-Bit Data Manual (MP014) for more detailed information].

The workspace-register files are nonoverlapping and contain 16 contiguous memory words. Each workspace register may hold data or addresses and function as operand registers, accumulators, address registers, or index registers (with the exception of R0). WP addresses in RAM will be one of four values: 8300_{16}, 8320_{16}, 8340_{16}, or 8360_{16}. For more information about the TMS9940, refer to the "TMS9940 16-Bit Microcomputer Data Manual".

The example design of Figure 6 shows how straightforward the system interconnections are. Note that only 11 signals connect the computer to the TMS9902. While five signals are used by the TMS9902 to communicate with a terminal, in this example a terminal is being used which is always clear to receive data when requested. Thus, the request to send ($\overline{RTS}$) and clear to send ($\overline{CTS}$) lines are tied together. The SN75189 line receiver converts the EIA plus and minus 12 volt signals to TTL levels as required by the TMS9902 while the SN75188 line driver converts from TTL to EIA levels as required by the terminal.

The display is a vacuum fluorescent display consisting of three basic electrode types enclosed in an evacuated glass chamber: filament (cathode), grid, and anode. There is one filament for the whole display, 20 grids corresponding to the 20 characters, and 700 anodes corresponding to the (35×20) dots.

The filament is heated to sufficient temperature to cause electrons to escape. When a positive voltage, with respect to the filament, is applied to a grid and a dot, the resultant electric field will accelerate the electrons toward the grid. Since the grid is a mesh, most of the electrons will pass the grid and be further accelerated toward the dot, colliding with the phosphorous before reaching the dot. The electrons convey most of their energy to the phosphorous causing light to be emitted (Figure 7).

To sufficiently accelerate the electrons, a positive voltage of about 30 volts is needed on the grids on the dots, so transistor drivers are used to buffer the TTL outputs. The inputs of these drivers are current limited by in-line 3300 ohm resistors and the outputs are pulled up to 30 volts through 3300 ohm resistors.

Turning off either a dot or a grid will reduce the flow of electrons enough to cause the corresponding dots to extinguish. But turning off the grids or dots alone does not produce a sufficient difference of potential to completely stop the flow of electrons in the display. If a 5 volt bias voltage is applied to the center tap of the filament transformer, a sufficient difference of potential can be developed to stop the flow of electrons when only the tube grids are turned off. Since a digit can be extinguished by turning off only its grid, the corresponding dots of each digit can be connected together and only 35 dot inputs and 20 grid inputs are needed to control all 700 dots. (Figure 7b.)

The inputs to the 35 dot drivers are the shift register outputs. In the processor, the ROM containing the character code is arranged so that 5 bits of each byte corresponds to one row of a character matrix. This is so each new row of the matrix code

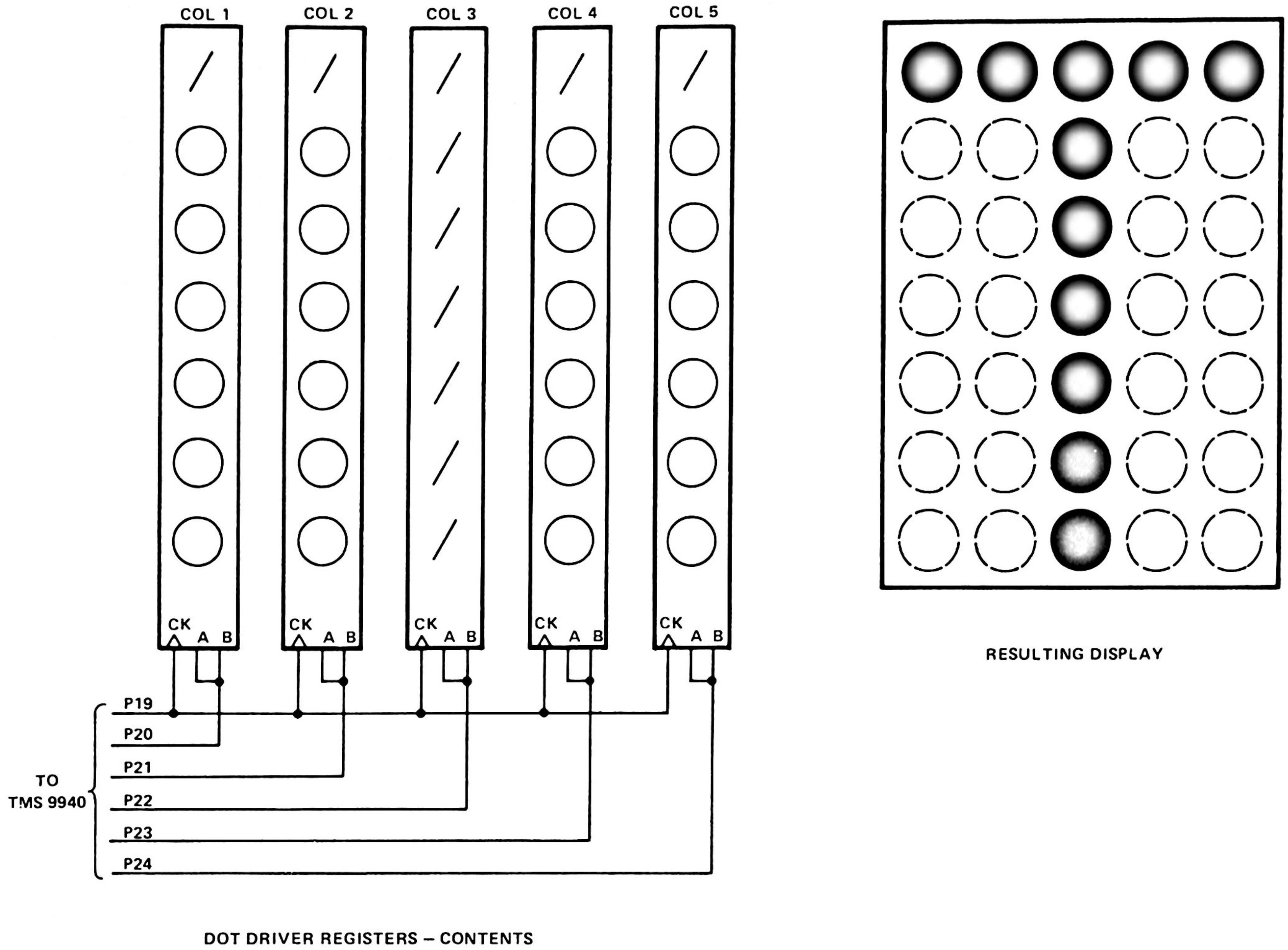

Figure 3. SIPO Registers/Display Correspondence

can be loaded to the TMS9940 output ports P20-P24 (the inputs to the five shift registers). I/O port P19 is then strobed to shift the data in until all seven rows of data have been shifted into the shift registers.

The TMS9940 must have the ability to turn on only one grid at a time and must be able to disable all grids before selecting the next character position (to prevent blurring). I/O ports P25-P29 are used to individually control each of the 20 grids. Three SN74S138, three-to-eight demultiplexers, decode these five I/O lines into 20 grid drivers. They are set up so the three least significant I/O lines control the select inputs to the de-mux's while the two most significant I/O lines (P28, P29) control the enable inputs. When P28, P29 equal 00_2, the first de-mux is enabled; when they equal 01_2, the second de-mux is enabled; when they equal 10_2, the third de-mux is enabled; and when they equal 11_2 all three de-mux's are disabled. Therefore, sequencing through the grids is simply a matter of incrementing the previous encoded grid code. All the grids are disabled by loading I/O ports P25-P29 with logical ones.

SYSTEM SOFTWARE

Throughout the following section it may be beneficial to refer to either the detailed program listing at the end of the report or the memory usage diagram of Figure 8. For specific information about the TMS9900 assembly language, refer to the "TMS9900 Microprocessor Assembly Language Programmers Guide".

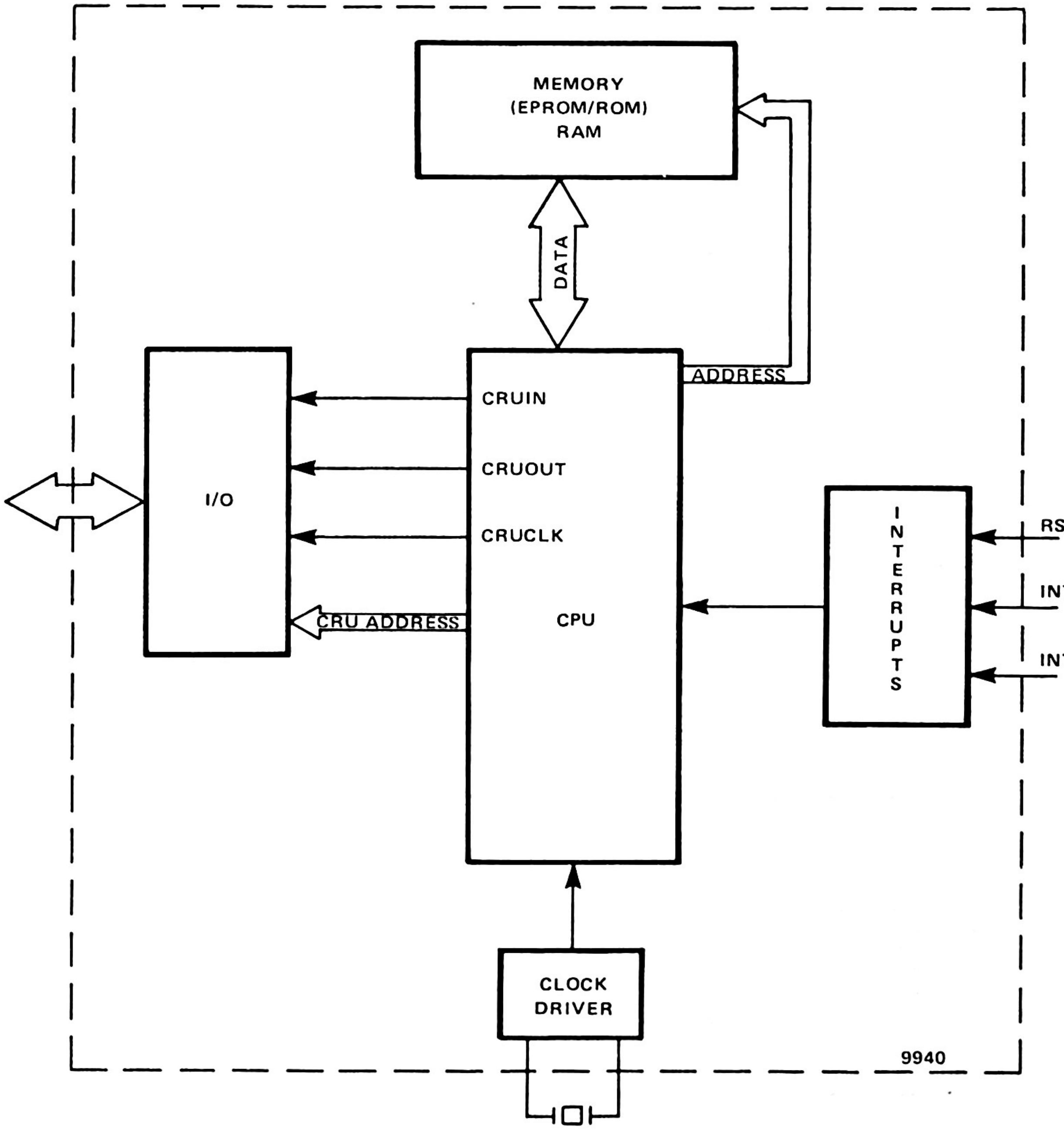

Figure 4. TMS9940 Simplified Block Diagram

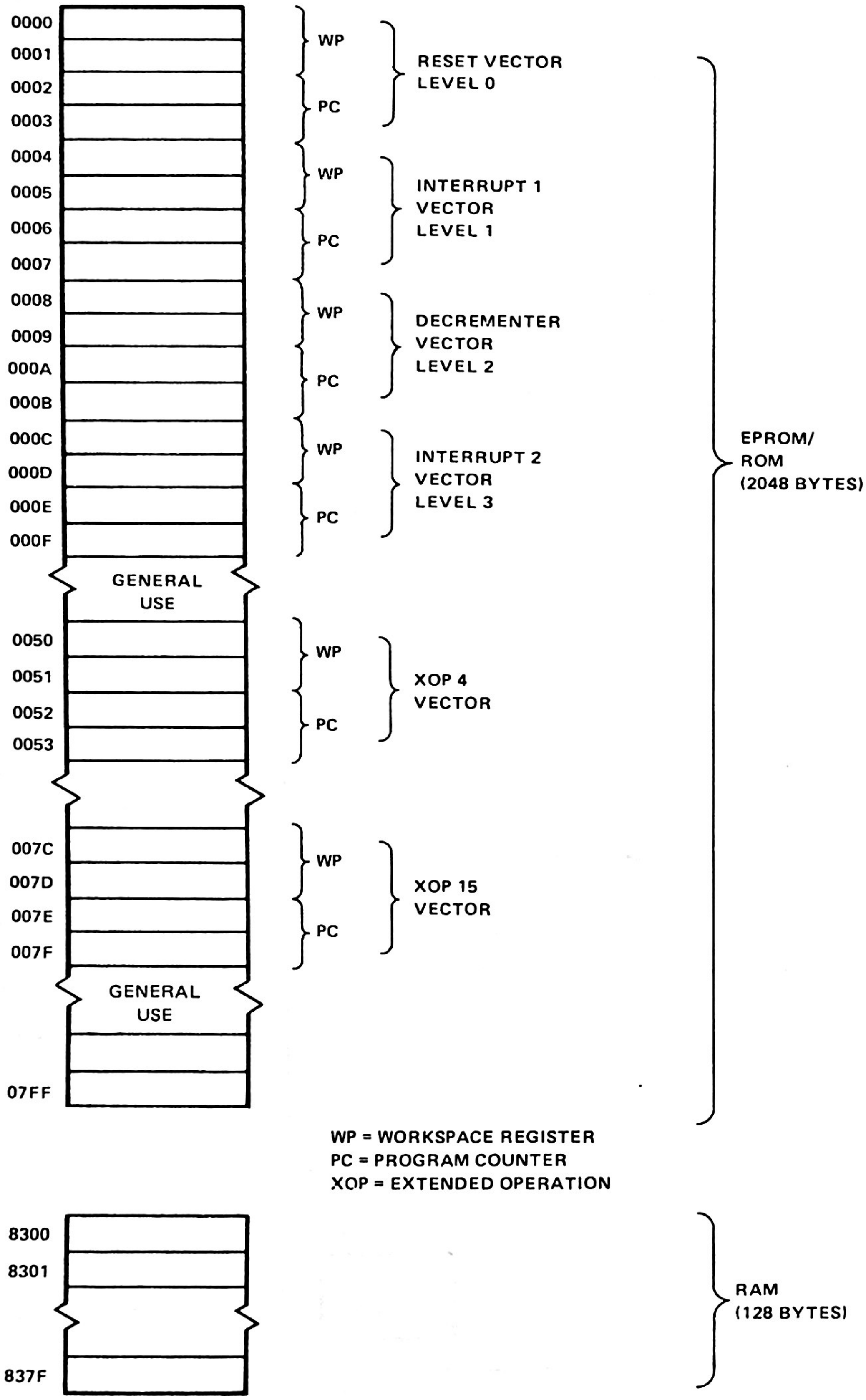

Figure 5. TMS9940 Memory Map

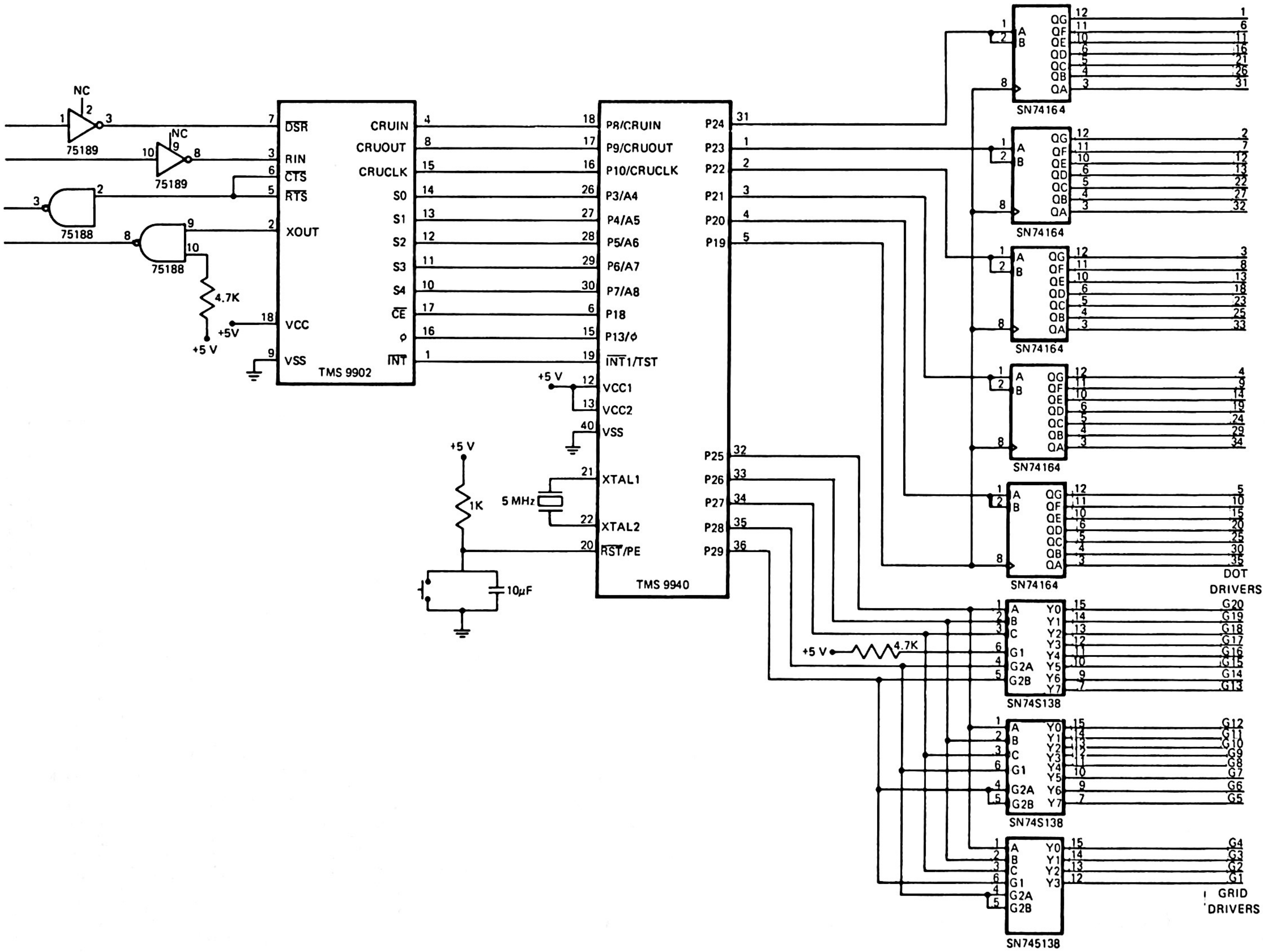

Figure 6. System Schematic

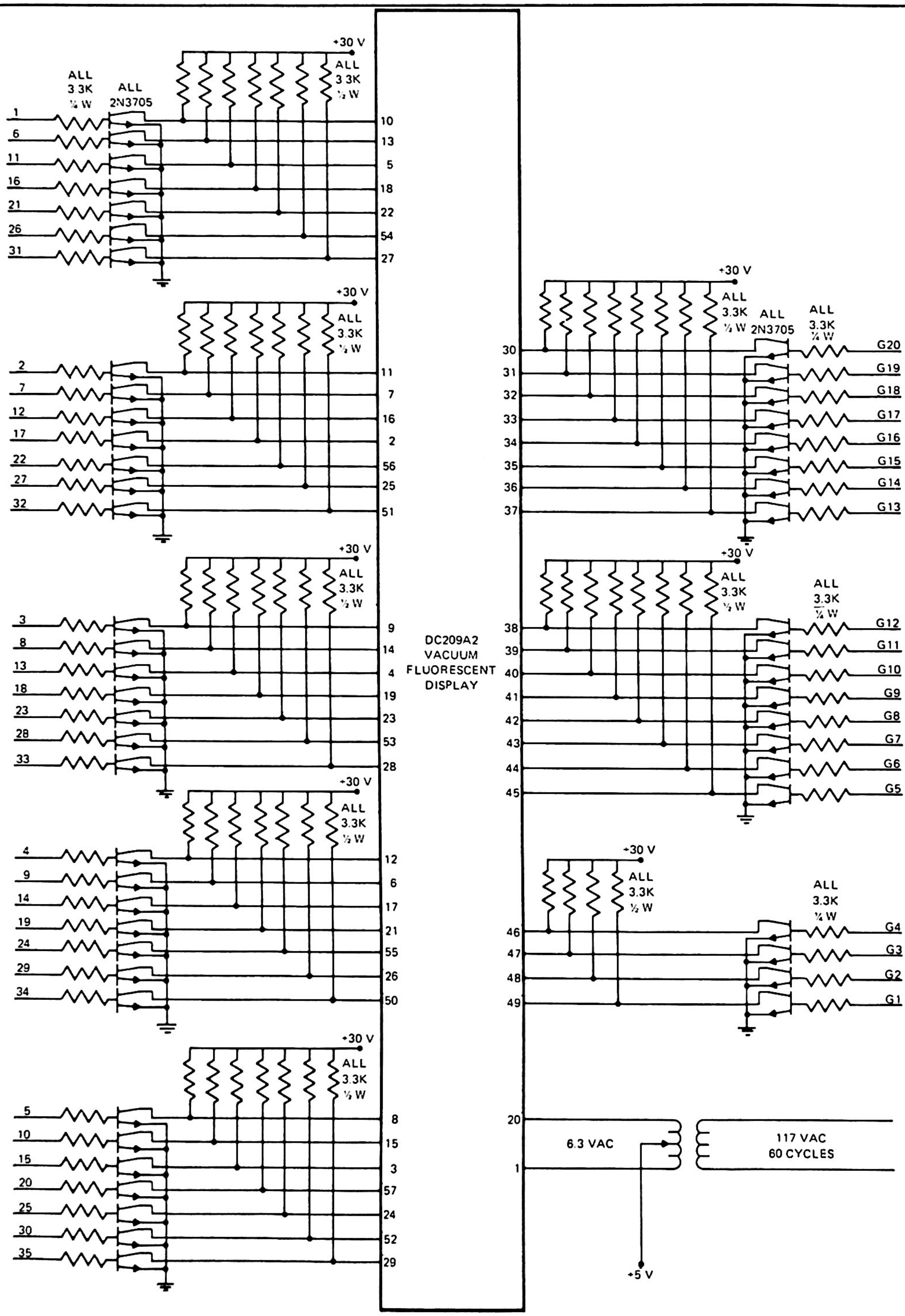

Figure 6. System Schematic (Continued)

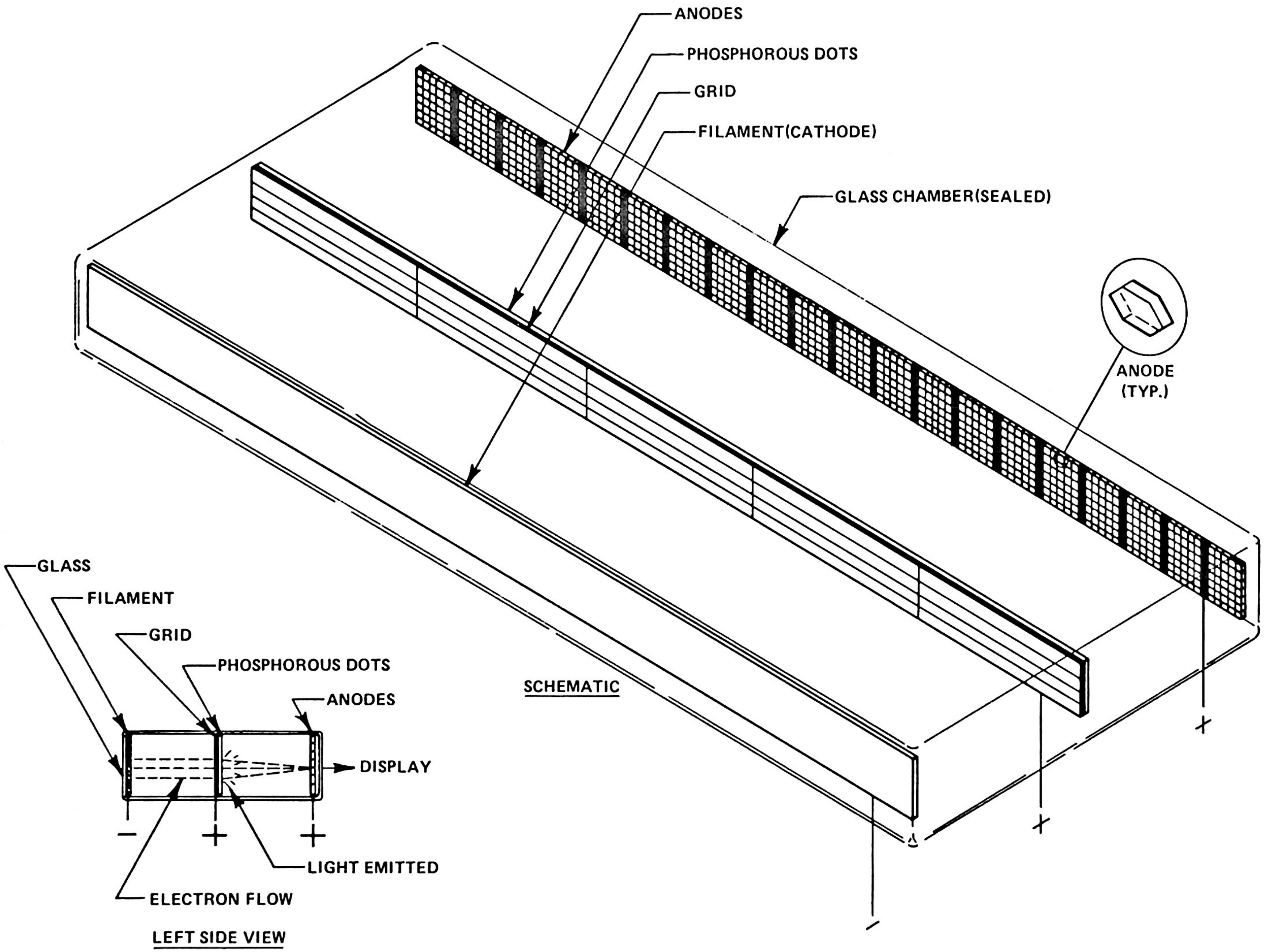

Figure 7a. Vacuum Fluorescent Display Interface

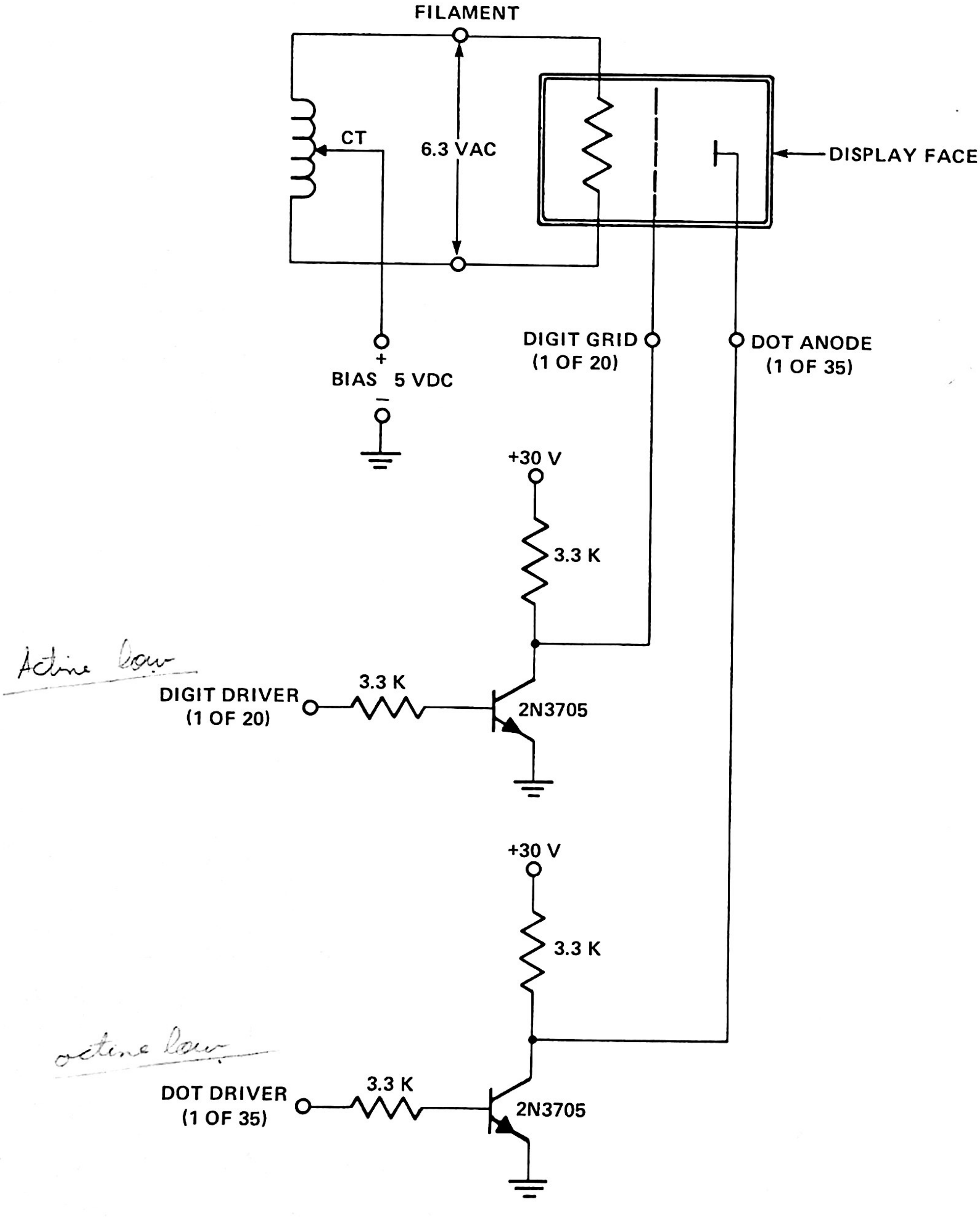

Figure 7b. Vacuum Fluorescent Display Interface

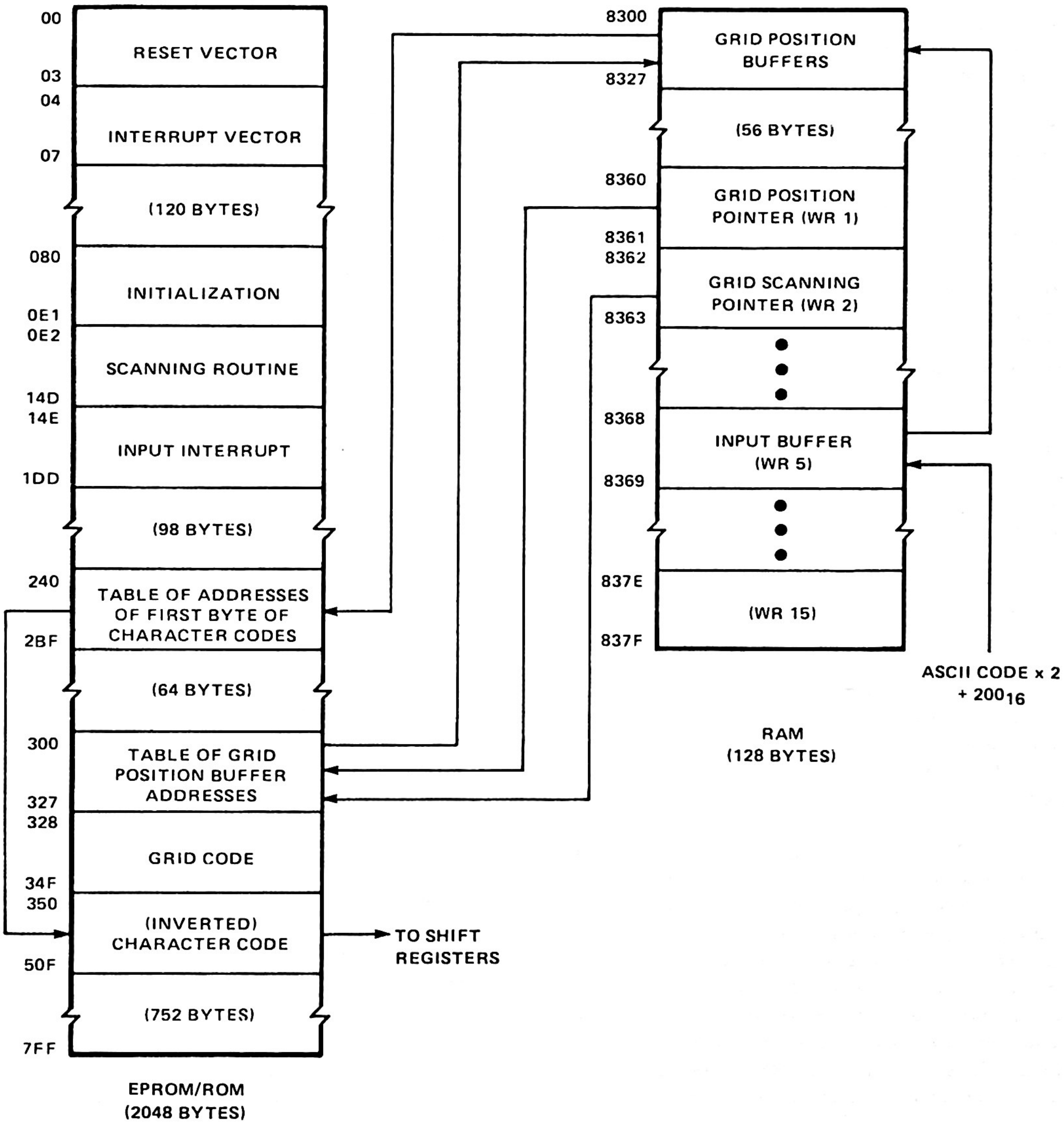

Figure 8. Display System Memory Usage Diagram

The memory usage diagram (Figure 8) shows the partitioning of read only and read/write memory with arrows helping to illustrate the functions of key memory sections from the time information (in the form of ASCII code) is received to the time it is to be displayed. Read only memory can be divided basically into three types: basic program routines, tables of addresses, and tables of code (for turning on appropriate grids and dots). Read/write memory is essentially either grid position buffers (which change when new display information is presented) or workspace registers.

Before examining the system software in some detail, it will be instructive to consider the general flow of information within the display system and the routines which control that flow (refer to the arrows in Figure 8). Since incoming ASCII data is received asynchronously and since the display must be synchronously refreshed (to provide a bright, uniform display) the scanning routine executes without interruption most of the time. The input interrupt routine is interrupt driven so that no matter when data is received (by the TMS9902), the scanning routine determines when it will be processed. This is accomplished by enabling an interrupt from the TMS9902 to invoke execution of the input interrupt routine. The scanning routine enables the interrupt only after extinguishing one character position and before illuminating the next so the ASCII data is processed without affecting display intensity.

Received ASCII data is examined for validity, multiplied by two, and added to 200_{16} in workspace register five (the input buffer) before being saved in one of the grid position buffers. The grid position buffers are 20 contiguous words of RAM telling the scanning routine which characters to display in the 20 character positions of the display. The grid position pointer (workspace register 1) points indirectly to the grid position buffer where the next modified ASCII code is to be saved.

The modified ASCII code in a grid position buffer is actually an address which points to one of 64 contiguous words in locations 240_{16} to $2BF_{16}$. The words in these locations are, in turn, addresses pointing to the first byte of a seven byte character code. These seven bytes are the inverted code output to the shift registers to display a character. Thus, the scanning routine inspects a grid position buffer, is pointed to a code, and uses that code to build one character in the display. The grid scanning pointer (workspace register 2), points indirectly to the current display character position.

The initialization routine configures the TMS9940 and the TMS9902. Program locations 80_{16} to 95_{16} primarily configure I/O ports P0-P10 of the TMS9940 for CRU expansion, while 96_{16} to $9F_{16}$ configure the clock and then the remaining ports as outputs. Program locations $A0_{16}$ to $A7_{16}$ set P18-P19 low and P20-29 high selecting the TMS9901 and blanking the display. Next, the grid position pointer and grid scanning pointer are set to digit 20 (location 300_{16}), the left-most digit of the display. The grid positions, which normally contain addresses of the code of the characters to be displayed, are all loaded with the address of the space character. Therefore, when the scan routine begins, the display will remain blank until a character is input.

Program locations $C6_{16}$ to $E1_{16}$ finish the initialization routine. The TMS9902 is first reset. Then the control register is loaded to select a character length of seven bits, even parity, and two stop bits for the transmitter. The receiver only tests for a single stop bit. Next, loading to the internal register is disabled by writing a logical zero to CRU bit 13. The next four instructions set the receive and transmit bit rates to 1200 BPS and disable the DSCINT, XINT, and TIMINT interrupts. Next, setting bit 18 of the CRU to logical one enables the RINT interrupt which occurs when the receive buffer is full.

Figure 9 is a flowchart of the scanning routine. The first sequence in the scan routine is to disable the grid drivers and enable the input interrupt. This turns off the previous character and allows any ASCII character received at this time to invoke the input interrupt routine. Next, as described earlier, the grid position buffer pointed to by the grid scanning pointer is used to point to the first byte of the corresponding character matrix code. Then, the least significant five bits of data from each of the seven matrix code bytes is loaded to the shift registers. Next, in program locations $12A_{16}$ to $12F_{16}$ the value of the grid scanning pointer plus 40 (28_{16}) is stored in a temporary register (WR 10). The register contents point to one of 20 grid codes residing in 328_{16} to $34F_{16}$. The input interrupt is now disabled and the five least significant bits of the grid code are output by the LDCR instruction (at 0134_{16}) to I/O lines P25-P29 causing the appropriate grid to turn on. The grid scanning pointer is checked to see if it is pointing to the right-most digit. If it is, the pointer is reset; if not, it is incremented. Note that in the flowchart as the grid scan pointer is incremented, the character position, called "N", decrements. A delay loop is inserted at the end of the scan routine to keep the digits scanned at a rate of approximately 100 Hertz. Somewhat slower scan rates produce flicker; somewhat faster scan rates cause the display to dim.

The input interrupt routine flowcharted in Figure 10 is entered when the TMS9902 issues an interrupt (signifying that a character has been received) and the scan routine has enabled the interrupt. The input interrupt routine always stores the received ASCII code in the input buffer and resets the RBRL interrupt flag inside the TMS9902 to prepare for the next ASCII character. The routine checks for one of six possible conditions: delete, a valid display character, carriage return, line feed, a backspace, or a forward space. If none of these conditions are found, this is an error condition and the routine simply returns to the scanning routine.

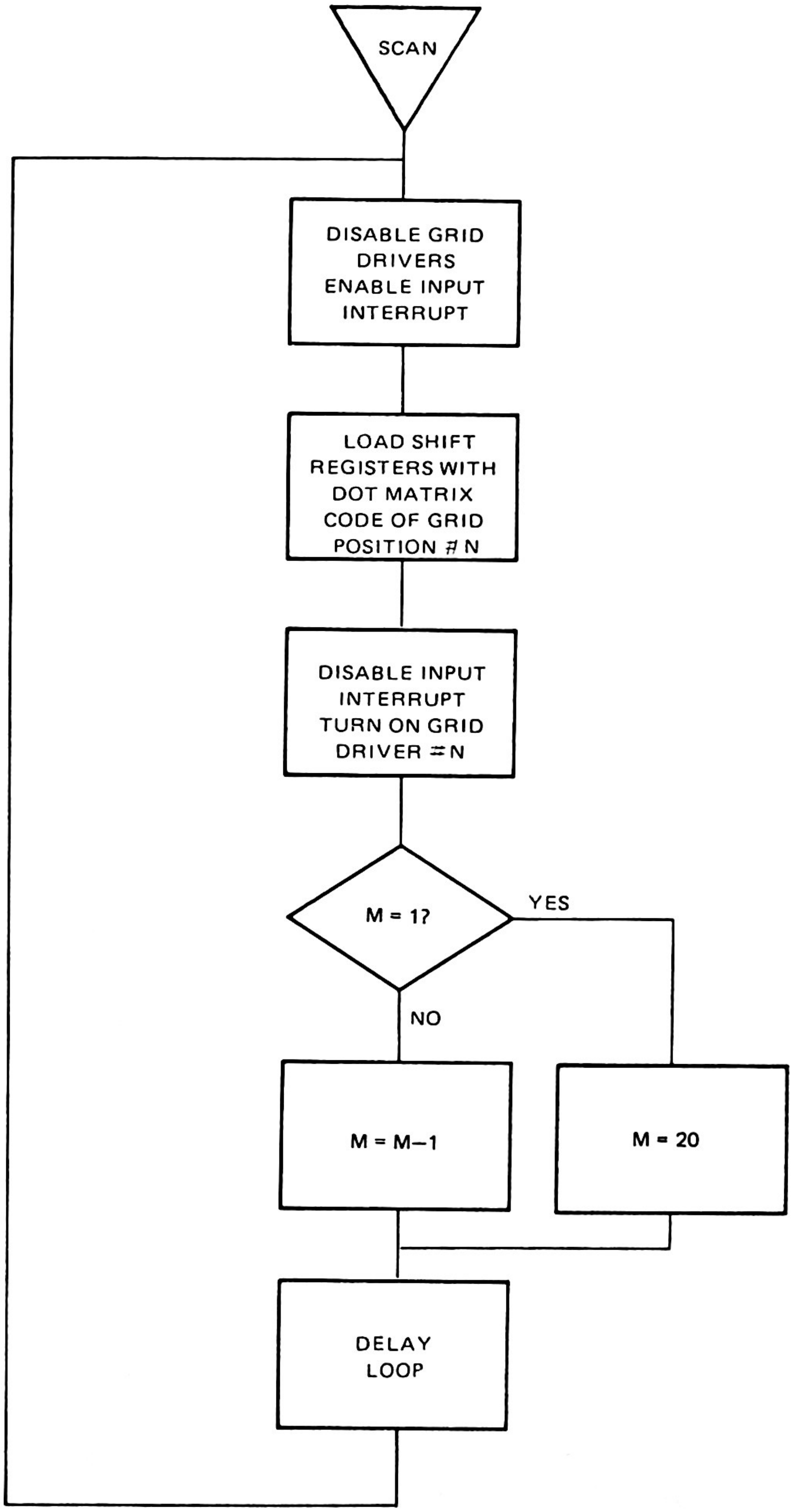

Figure 9. Scanning Routine

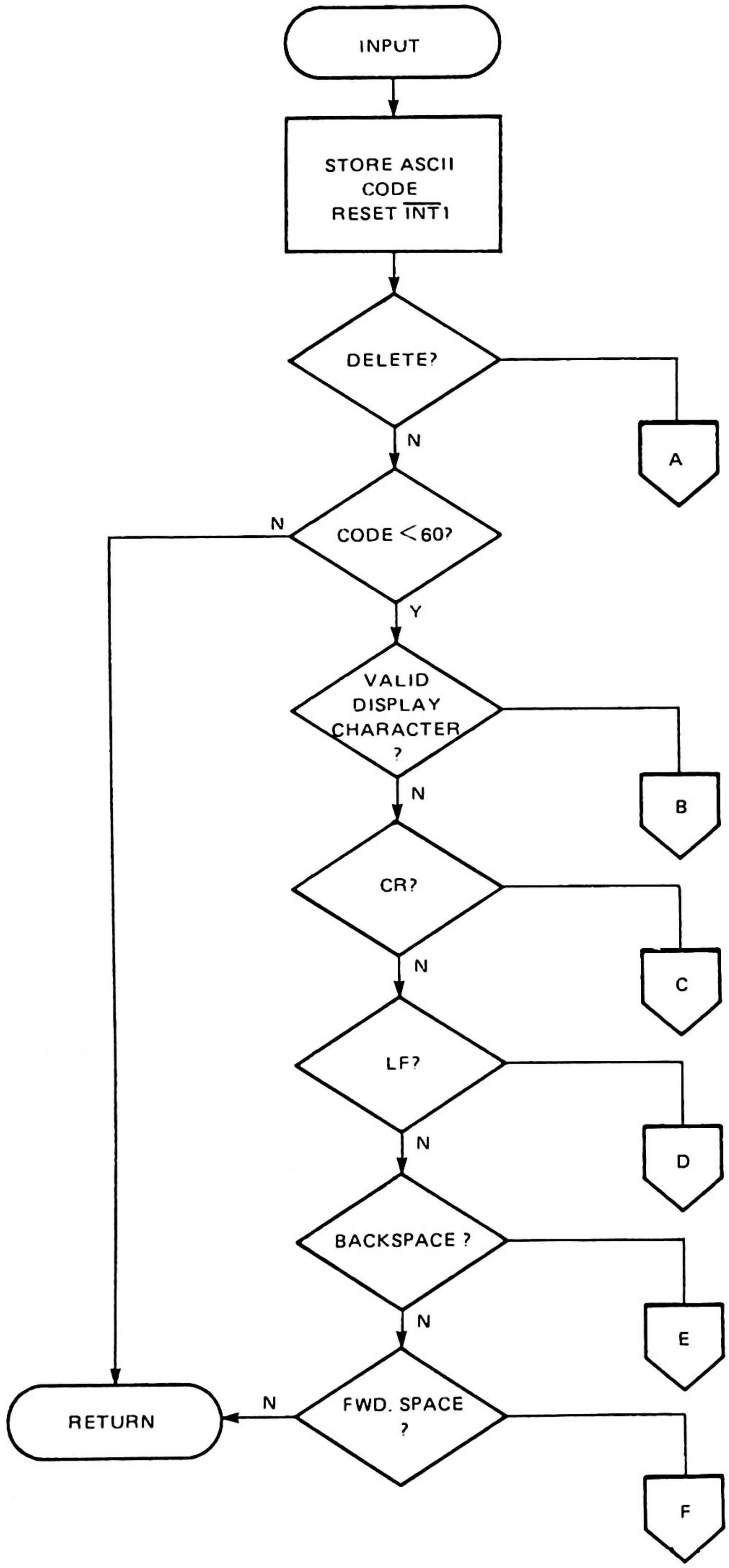

Figure 10. Input Interrupt Routine (Sheet 1 of 2)

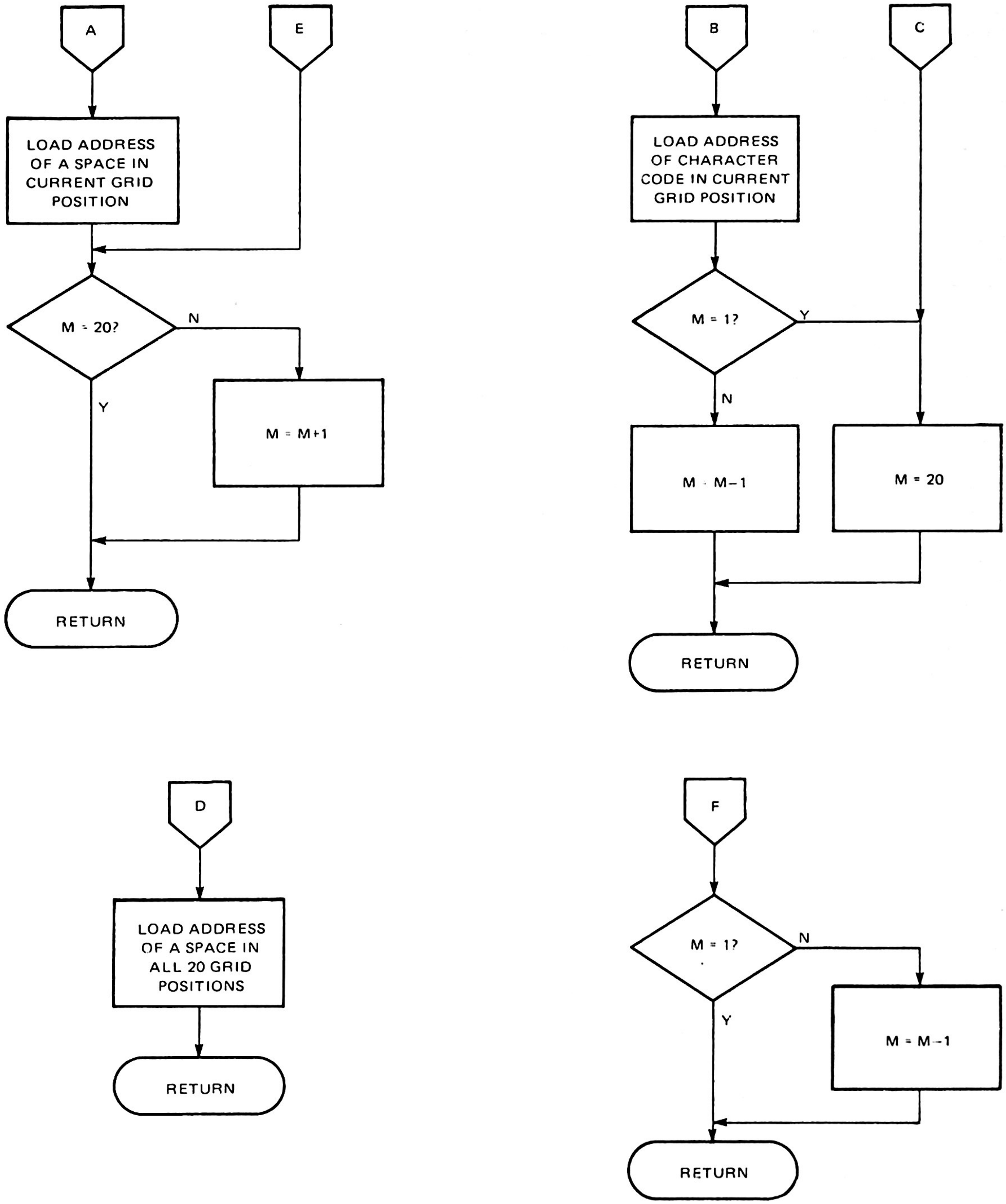

Figure 10. Input Interrupt Routine (Sheet 2 of 2)

A delete loads the current grid position buffer with the address of a space and decrements the grid position pointer. If a valid display character has been received, the address of the matrix code for that character is calculated (ASCII code $\times$ 2 + 200_{16}), and loaded into the present grid position buffer. The grid position pointer is checked to see if it is all the way to the right-most character position (M = 1 in the diagram). If it is, the pointer is reset to its initial value (M = 20). If it is not, the pointer is incremented (M is decremeted).

A carriage return resets the grid position pointer to the left-most grid while a line feed loads all grid positions with the address for a space. A backspace decrements the grid position pointer (increments M) if it is not at the left-most character position. A forward space increments the grid position pointer (decrements M) if it is not at the right-most character position. In all cases the input interrupt routine returns to the scanning routine.

The character code memory is given in Table 1. As explained earlier, a simple method is used to calculate the address of a character code. By multiplying the received ASCII code (20_{16} to $5F_{16}$) by two and adding to the result 200_{16}, the addresses 240_{16} to $2BE_{16}$ are calculated, which, in turn, contain the addresses of the first bytes of the seven byte character codes. The inverted character code is in memory locations 350_{16} to $50F_{16}$.

For example, suppose the character "T" is typed. Its ASCII code is 54_{16}, which multiplied by two gives $A8_{16}$. Adding that to 200_{16} gives $2A8_{16}$. The address at this memory location is $4BC_{16}$ (Table 1). The seven bytes starting at $4BC_{16}$ are EO,FB,FB,FB,FB,FB,FB which are the inverted code output to the shift registers to display the character "T" (reference Figure 3).

CONCLUSION

An example design has been presented demonstrating the TMS9940 microcomputer and a 20 position vacuum fluoresent display tube combined to give a powerful, cost-effective, 20 character display device. A comparable TTL implementation of the same device could easily involve three times the number of integrated circuits and have much less flexibility in terms of communications handling, redefinition of character fonts, self testing diagnostics, or data processing, etc. The power consumption of the TTL implementation would probably be on the order of two times greater; the space consumed could also easily be doubled; the cost to build, troubleshoot, and repair would be significantly greater; while the reliability would be less.

The display system presented offers compact circuitry, simplified communications handling, user definable character fonts, and with some imagination, data checking, data modification, multiple display buffering, or special display control such as scrolling or selective flashing. Sufficient hardware details have been given to easily begin a working model and a detailed software listing with thorough commenting follows.

Table 1. Character Code Memory

Base Address +0	+2	+4	+6	+8	+A	+C	+E
0240=0350	0357	035E	0365	036C	0373	037A	0381
0250=0388	038F	0396	039D	03A4	03AB	03B2	03B9
0260=03C0	03C7	03CE	03D5	03DC	03E3	03EA	03F1
0270=03F8	03FF	0406	040D	0414	041B	0422	0429
0280=0430	0437	043E	0445	044C	0453	045A	0461
0290=0468	046F	0476	047D	0484	048B	0492	0499
02A0=04A0	04A7	04AE	04B5	04BC	04C3	04CA	04D1
02B0=04D8	04DF	04E6	04ED	04F4	04FB	0502	0509
02C0=0000	0000	0008	0000	0000	0000	0000	0000
02D0=0000	0000	0000	0000	0000	0000	0000	0000
02E0=0000	0000	0000	0000	0000	0000	0000	0000
02F0=0000	0000	0000	0000	0000	0000	0000	0000
0300=8300	8302	8304	8306	8308	830A	830C	830E
0310=8310	8312	8314	8316	8318	831A	831C	831E
0320=8320	8322	8324	8326	0000	0100	0200	0300
0330=0400	0500	0600	0700	0800	0900	0A00	0B00
0340=0C00	0D00	0E00	0F00	1000	1100	1200	1300
0350=FFFF	FFFF	FFFF	FFF3	F3F3	F3FF	F3F3	F5F5
0360=FFFF	FFFF	FFFF	F5E4	FFE4	F5FF	F1EA	EBF1
0370=FAEA	F1E7	E6FD	FBF7	ECFC	F7EB	EBF7	EAED
0380=F2FD	FBF7	FFFF	FFFF	FDFB	F7F7	F7FB	FDF7
0390=FBFD	FDFD	FBF7	FFFB	F1E0	F1FB	FFFF	FBFB
03A0=E0FB	FBFF	FFFF	FFF3	F3FB	F7FF	FFFF	E0FF
03B0=FFFF	FFFF	FFFF	FFF3	F3FF	FEFD	FBF7	EFFF
03C0=F1EE	EEEE	EEEE	F1FB	F3FB	FBFB	FBF1	F1EE
03D0=FEF1	EFEF	E0F1	EEFE	F9FE	EEF1	FDF9	F5ED
03E0=E0FD	FDE0	EFE1	FEFE	EEF1	FBF7	EFE1	EEEE
03F0=F1E0	FEFD	FBF7	F7F7	F1EE	EEF1	EEEE	F1F1
0400=EEEE	F0FE	FDF3	FFF3	F3FF	F3F3	FFF3	F3FF
0410=F3F3	FBF7	FDFB	F7EF	F7FB	FDFF	FFE0	FFE0
0420=FFFF	F7FB	FDFE	FDFB	F7F1	EEFD	FBFB	FFFB
0430=F1EE	FEF2	EAE9	F3F1	EEEE	E0EE	EEEE	E1F6
0440=F6F1	F6F6	E1F1	EEEF	EFEF	EEF1	E1F6	F6F6
0450=F6F6	E1E0	EFEF	E1EF	EFE0	E0EF	EFE1	EFEF
0460=EFF0	EFEF	E9EE	EEF1	EEEE	EEE0	EEEE	EEF1
0470=FBFB	FBFB	FBF1	FEFE	FEFE	FEEE	F1EE	EDEB
0480=E7EB	EDEE	EFEF	EFEF	EFEF	E0EE	E4EA	EAEE
0490=EEEE	EEE6	EAEC	EEEE	EEE0	EEEE	EEEE	EEE0
04A0=E1EE	EEE1	EFEF	EFF1	EEEE	EEEA	EDF2	E1EE
04B0=EEE1	EBED	EEF1	EEF7	FBFD	EEF1	E0FB	FBFB
04C0=FBFB	FBEE	EEEE	EEEE	EEF1	EEEE	EEF5	F5FB
04D0=FBEE	EEEE	EAEA	EAF5	EEEE	F5FB	F5EE	EEEE
04E0=EEF5	FBFB	FBFB	E0FE	FDFB	F7EF	E0E3	EFEF
04F0=EFEF	EFE3	FFF5	FFFB	EEF1	FFF8	FEFE	FEFE
0500=FEF8	FBF5	EEFF	FFFF	FFFF	FFFF	FFFF	FFE0

```
                      IDT    'VACUUM'
                     ***********************************************************
                     *             TMS9940 VACUUM-FLUORESCENT DISPLAY APPLICATION
                     *                         SOFTWARE  MONITOR
                     *                                          6/14/78
                     *
                     *
                     ***********************************************************
                     *       CRU BASE ADDRESSES
                     ***********************************************************
          02C0 *DS9901 EQU   >2C0              TMS990/40DS-TMS9901
          0306  IOCONF EQU   >306              I/O CONFIGURATION
          03A4  IODIRC EQU   >3A4              I/O DIRECTION
          03E4  IODAT  EQU   >3E4              I/O DATA
          03F2  GRIDDR EQU   >3F2              GRID DRIVER I/O
          03E8  SHIFTR EQU   >3E8              SHIFT REGISTER I/O
                ***********************************************************
                *       WORKSPACE REGISTERS
                ***********************************************************
          0001  GRIDPP EQU   1                 GRID POSITION POINTER
          0002  GRIDSP EQU   2                 GRID SCANNING POINTER
          0003  GRBIN1 EQU   3                 GRID BUFFER INITIALIZER 1
          0004  GRBIN2 EQU   4                 GRID BUFFER INITIALIZER 2
          0005  INPBUF EQU   5                 INPUT BUFFER
          0006  GRDRCN EQU   6                 GRID DRIVER CONTROL
          0007  DTCOUT EQU   7                 DOT CODE COUNTER
          0008  COUNTR EQU   8                 COUNTER
          0009  TEMP2  EQU   9                 TEMPORARY REGISTER
          000A  TEMP1  EQU   10                TEMPORARY REGISTER
          000C  CRUBAS EQU   12                CRU BASE REGISTER
                ***********************************************************
                *       CONSTANT INITIALIZATIONS
                ***********************************************************
          0300  GRPTIN EQU   >300              GRID POINTER INITIALIZER
          8300  DOTBUF EQU   >8300             1ST DOT BUFFER
          0028  DTBFCT EQU   >28               DOT BUFFER COUNT
          0240  SPACHR EQU   >240              SPACE CHARACTER
          6200  CTRL   EQU   >6200             TMS9902 CONTROL REGISTER
          01A0**BAUDRT EQU   >1A0              BAUD RATE
          0326  GRID1  EQU   >326              GRID #1
          0200  CHADST EQU   >200              CHARACTER ADDRESS SETUP
                ***********************************************************
                *       BEGIN PROGRAM
                ***********************************************************
     0000 8360  BEGIN  DATA >8360              RESET WORKSPACE POINTER
     0002 0080'        DATA START              RESET PROGRAM COUNTER
     0004 8360         DATA >8360              INTERRUPT #1 WORKSPACE POINT.
     0006 014E'        DATA INPINT             INTERRUPT #1 PROGRAM COUNTER
     0080       START  BES  >78                PROGRAM STARTS @>80
                ***********************************************************
                *       INITIALIZATION
                ***********************************************************
     0080 02E0         LWPI >8360              SET UP WORKSPACE POINTER
     0082 8360
     0084 020C *       LI   CRUBAS,DS9901      SET UP CRU BASE TO TMS9901
     0086 02C0
     0088 020A *       LI   TEMP1,>E00         SET UP INTERRUPT INITIALIZE
     008A 0E00
     008C 310A *       LDCR TEMP1,4            ENABLE INTERRUPTS 1-3
     008E 020C         LI   CRUBAS,IOCONF      POSITION CRU FOR CONFIG. I/O
```

```
0090 0306
0092 020A              LI    TEMP1,>500         CONFIGURE P0-P10 FOR CRU EXP.
0094 0500
0096 310A              LDCR  TEMP1,4            CONFIG. P11-P12, PHI, P14-P16
0098 020C              LI    CRUBAS,IODIRC      POSITION CRU FOR I/O DIREC.
009A 03A4
009C 070A              SETO  TEMP1
009E 330A              LDCR  TEMP1,12           SET P18-P31 TO OUTPUTS
00A0 020C              LI    CRUBAS,IODAT       POSITION CRU FOR I/O DATA
00A2 03E4
00A4 020A              LI    TEMP1,>FFC         P18-P19 LOW, P20-P29 HIGH
00A6 0FFC
00A8 330A              LDCR  TEMP1,12           ENABLE TMS9902, SET P19-P29
00AA 0201              LI    GRIDPP,GRPTIN      SET GRID POSITION POINTER
00AC 0300
00AE 0202              LI    GRIDSP,GRPTIN      SET GRID SCANNING POINTER
00B0 0300
00B2 0203              LI    GRBIN1,DOTBUF      LOAD THE
00B4 8300
00B6 0208              LI    COUNTR,DTBFCT      DOT BUFFERS
00B8 0028
00BA 0204              LI    GRBIN2,SPACHR      WITH SPACES
00BC 0240
00BE C4C4    CLBF      MOV   GRBIN2,*GRBIN1
00C0 05C3              INCT  GRBIN1
00C2 0648              DECT  COUNTR
00C4 16FC              JNE   CLBF
00C6 04CC              CLR   CRUBAS             POSITION CRU FOR TMS9902
00C8 020A              LI    TEMP1,CTRL         SET UP CONTROL REGISTER
00CA 6200
00CC 0209              LI    TEMP2,BAUDRT       1200 BAUD RATE
00CE 01A0
00D0 1D1F              SBO   31                 RESET TMS9902
00D2 320A              LDCR  TEMP1,8            LOAD CONT. REG., RST. LDCTRL
00D4 1E0D              SBZ   13                 RESET INTERVAL REGISTER
00D6 3309              LDCR  TEMP2,12           SET REC. AND TRANS. @300 BAUD
00D8 1E15              SBZ   21                 DISABLE DSCH INTERRUPT
00DA 1E14              SBZ   20                 DISABLE TIMELP INTERRUPT
00DC 1E13              SBZ   19                 DISABLE XBIEND INTERRUPT
00DE 1D12              SBO   18                 ENABLE RIENB INTERRUPT
00E0 0588              INC   COUNTR             INITIALIZE COUNTR
                       ****************************************************************
                       *       SCAN ROUTINE
                       ****************************************************************
00E2 020C    SCAN      LI    CRUBAS,GRIDDR      POSITION CRU TO GRID DRIVERS
00E4 03F2
00E6 0706              SETO  GRDRCN             DISABLE GRID DRIVERS
00E8 3146              LDCR  GRDRCN,5
00EA 0300              LIMI  1                  ENABLE INPUT INTERRUPT
00EC 0001
00EE 020C              LI    CRUBAS,SHIFTR      POSITION CRU TO SHIFT REG.
00F0 03E8
00F2 C0D2              MOV   *GRIDSP,GRBIN1     LOAD GRBIN1 WITH ADDR.OF DOT
00F4 C113              MOV   *GRBIN1,GRBIN2     CODE OF 1ST ROW OF CHAR.PNTED
00F6 C0D4              MOV   *GRBIN2,GRBIN1     TO BY THE GRID SCAN. POINTER
00F8 0207              LI    DTCOUT,>7          SET UP COUNTER FOR 7 LOOPS
00FA 0007
00FC 8042    LDSHR     C     GRIDSP,GRIDPP      CHECK FOR CURSOR
00FE 160D              JNE   LDSHRO             NOT AT CURSOR, GO TO LDSHRO
0100 0608              DEC   COUNTR
```

```
                      SDSMAC      3.1 *     09:15:28 FRIDAY, JUN 16, 1978.
FLUORESCENT DISPLAY APPLICATION                                              PAGE 0006

 0197                ******************************************************************
 0198                *       FORWARD SPACE
 0199                *
 0200 01D4 0281      FWDSPC CI    GRIDPP,GRID1         CHECK IF RIGHT MOST GRID
      01D6 0326
 0201 01D8 1301             JEQ   FWDSTP
 0202 01DA 05C1             INCT  GRIDPP               IF NOT, INCREMENT GRID POINT.
 0203 01DC 0380      FWDSTP RTWP
 0204                       END
NO ERRORS
```

*These lines are omitted when using the TMS9940, they are included when using the TM990/40DS Development System.

**These rates change depending on the internal clock rate.

TEXAS INSTRUMENTS
INCORPORATED

SEMICONDUCTOR GROUP

POST OFFICE BOX 225012 • DALLAS, TEXAS 75265

TI Worldwide Sales Offices

ALABAMA: **Huntsville**, 4717 University Dr., Suite 101, Huntsville, AL 35805, (205) 837-7530.

ARIZONA: **Phoenix**, P.O. Box 35160, 8102 N. 23rd Ave., Suite A, Phoenix, AZ 85069, (602) 249-1313.

CALIFORNIA: **Costa Mesa**, 3186J Airway, Costa Mesa, CA 92626, (714) 540-7311. **El Segundo**, 831 S. Douglas St., El Segundo, CA 90245, (213) 973-2571. **San Diego**, 4333 View Ridge Ave., Suite B., San Diego, CA 92123, (714) 278-9600. **Sunnyvale**, P.O. Box 9064, 776 Palomar Ave., Sunnyvale, CA 94086, (408) 732-1840.

COLORADO: **Denver**, 9725 E. Hampden St., Suite 301, Denver, CO 80231, (303) 751-1780.

CONNECTICUT: **Hamden**, 2405 Whitney Ave., Hamden, CT 06518, (203) 281-0074.

FLORIDA: **Clearwater**, 2280 U.S. Hwy. 19 N., Suite 232, Clearwater, FL 33515, (813) 725-1861. **Ft. Lauderdale**, 4600 W. Commercial Blvd., Ft. Lauderdale, FL 33319, (305) 733-3300. **Winter Park**, 1850 Lee Rd., Suite 115, Winter Park, FL 32789, (305) 644-3535.

GEORGIA: **Atlanta**, 330 Northeast Expy., Suite 9, Atlanta, GA 30341, (404) 458-3140.

ILLINOIS: **Arlington Heights**, 515 W. Algonquin, Arlington Heights, IL 60005, (312) 640-3000.

INDIANA: **Ft. Wayne**, 2020 Inwood Dr., Ft. Wayne, IN 46805, (219) 424-5174. **Indianapolis**, 5726 Professional Cir., Suite 103, Indianapolis, IN 46241, (317) 248-8555.

MASSACHUSETTS: **Waltham**, 504 Totten Pond Rd., Waltham, MA 02154, (617) 890-7400.

MICHIGAN: **Southfield**, Central Park Plaza, 26211 Central Park Blvd., Suite 215, Southfield, MI 48076, (313) 353-0830.

MINNESOTA: **Edina**, A.I.C. Bldg., Suite 202, 7615 Metro Blvd., Edina, MN 55435, (612) 835-2900.

MISSOURI: **Kansas City**, 8080 Ward Pkwy., Kansas City, MO 64114, (816) 523-2500. **St. Louis**, 2368 Schuetz Rd., St. Louis, MO 63141, (314) 569-0801.

NEW JERSEY: **Clark**, 1245 Westfield Ave., Clark, NJ 07066, (201) 574-9800.

NEW MEXICO: **Albuquerque**, 5907 Alice N.E., Suite E, Albuquerque, NM 87110, (505) 265-8491.

NEW YORK: **East Syracuse**, 6700 Old Collamer Rd., East Syracuse, NY 13057, (315) 463-9291. **Endicott**, 112 Nanticoke Ave., P.O. Box 618, Endicott, NY 13760, (607) 754-3900. **Melville**, 1 Huntington Quadrangle, Suite 1C01, Melville, NY 11746, (516) 293-2560. **Poughkeepsie**, 201 South Ave., Poughkeepsie, NY 12601, (914) 473-2900. **Rochester**, 1210 Jefferson Rd., Rochester, NY 14623, (716) 461-1800.

NORTH CAROLINA: **Charlotte**, 1 Woodlawn Green, Woodlawn Rd., Charlotte, NC 28210, (704) 527-0930. **Raleigh**, 4130 C 2 Camelot Dr., Raleigh, NC 27609, (919) 787-9376.

OHIO: **Cleveland**, Belmont Bldg., Suite 120, 28790 Chagrin Blvd., Cleveland, OH 44122, (216) 464-2990. **Dayton**, Kingsley Bldg., 4124 Linden Ave., Dayton, OH 45432, (513) 253-3121.

OKLAHOMA: **Tulsa**, 3105 E. Skelly Dr., Suite 110, Tulsa, OK 74128, (918) 749-9548.

OREGON: **Beaverton**, 10700 S.W. Beaverton Hwy., Suite 565, Beaverton, OR 97005, (503) 643-6759.

PENNSYLVANIA: **Ft. Washington**, 275 Commerce Dr., Suite 300, Ft. Washington, PA 19034, (215) 643-6450.

TENNESSEE: **Johnson City**, P.O. Drawer 1255, Erwin Hwy., Johnson City, TN 37601, (615) 926-1167.

TEXAS: **Dallas**, 6000 Denton Dr., P.O. Box 225012, M/S 366, Dallas, TX 75265, (214) 238-6805. **Houston**, 9000 Southwest Frwy., Suite 400, Houston, TX 77036, (713) 776-6511.

VIRGINIA: **Arlington**, Crystal Square 4, 1745 Jefferson Davis Hwy., Suite 600, Arlington, VA 22202, (703) 553-2200. **Richmond**, 3930 Beulah Rd., Richmond, VA 23234, (804) 275-8148.

WASHINGTON: **Bellevue**, 700 112th N.E., Suite 10, Bellevue, WA 98004, (206) 455-3480.

CANADA: **St. Laurent**, 945 McCaffery St., St. Laurent H4T1N3, Quebec, Canada, (514) 341-3232. **Richmond Hill**, 280 Centre St. E., Richmond Hill L4C1B1, Ontario, Canada, (416) 884-9181. A

ARGENTINA, Texas Instruments Argentina S.A.I.C.F. Km. 25, 5 Ruta Panamericana Don Torcuato, C.C. 2296, 1000-Correo Central, Buenos Aires, Argentina, 748-1141.

ASIA, Texas Instruments Asia Ltd. 902, Asian House, 1, Hennessy Rd., Hong Kong, 05-279041. P.O. Box 2093, 990 Bendemeer Rd., Singapore 1, Republic of Singapore, 65-2581122. Room 507, Chia Hsin Bldg., 96 Chung Shan North Rd., Sec. 2, Taipei, Taiwan, 02-5630731. Aoyama Tower Bldg., 4, 5, & 6F, 24-15 Minami Aoyama, 2-Chome, Minato-Ku., Tokyo, Japan 107, 03-402-6171.

AUSTRALIA, Texas Instruments Australia Ltd. Unit 1A, 5 Byfield St., P.O. Box 106, North Ryde, N.S.W. 2113, Sydney, Australia, 02-887-1122.

AUSTRIA, Texas Instruments Ges. m.b.H. Rennweg 17, 1030 Vienna, Austria, 0222-724186.

BELGIUM, Texas Instruments S/A Mercurie Centre, Raketstraat 100, Rue De La Fusee, 1130 Brussels, Belgium, 02-7208000.

BRAZIL, Texas Instrumentos Electronicos do Brasil Ltda. Rua Padre Pereira De Androde, 591 Cep-05469 Sao Paulo, Brazil, 011-260-6347.

DENMARK, Texas Instruments A/S Marielundvej 46E, 2730 Herlev, Denmark, 02-917400.

FINLAND, Texas Instruments Finland OY Fressenkatu 6, P.L. 917, 00101 Helsinki 10, Finland, 80-408300.

FRANCE, Texas Instruments France La Boursidiere, Bloc C, R.N. 186, 92350 Le Plessis Robinson, France, 01-6303243. 31 Quai Rambaud, 69002 Lyon, France, 078-373585. 1, Av. de la Chartreuse, 38240 Meylan, France, 076-904574. 9, Place de Bretange, 35000 Rennes, France, 099-795481. 100-102 Alle de Barcelone, Residence L'Autay, 31000 Toulouse, France, 061-213032.

GERMANY, Texas Instruments Deutschland GmbH: Kurfuerstendamm 146, 1000 Berlin 31, Germany, 030-8927063. Lazarettstrasse 19, 4300 Essen, Germany, 0201-233551. Akazienstrasse 22-26, 6230 Frankfurt-Griesheim, Germany, 0611-399061. Haggertystrasse 1, 8050 Freising, Germany, 08161-801. Riethorst 4, 3000 Hanover 51, Germany, 0511-648021. Frankfurter Ring 243, 8000 Munich 40, Germany, 089-325011. Krefelderstrasse 11-15, 7000 Stuttgart 50, Germany, 0711-547001.

ITALY, Texas Instruments Italia Spa. Via Europa 38/44, Cologno Monzese, Milan, Italy, 02-253-2451. Via Salaria 1319, 00138 Rome, Italy, 06-6917127. Via Montebello 27, 10124 Turin, Italy, 011-832276.

MEXICO, Texas Instruments de Mexico S.A. Poniente 116 #489, Col. Industrial Vallejo, Mexico City 15, D.F., Mexico, 905-567-9200.

NETHERLANDS, Texas Instruments Holland BV. Laan Van de Helende Meesters 421 A, P.O. Box 283, Amstelveen, Holland, 020-473391.

NORWAY, Texas Instruments A/S Ryensvingen 15, Oslo 6, Norway, 02-689487.

PORTUGAL, Texas Instruments Equipamento Electronico LDA Rua Eng. Frederico Ulrich, 2650 Moreira Da Maia, Portugal, 948-1003.

SPAIN, Texas Instruments Espana S.A. Calle Mallorca 272-276, 12 Barcelona 12, Spain, 03-2152950.

SWEDEN, Texas Instruments International Trade Corporation (Sverigefilialen) Norra Hannvagen 3, Fack S-100 54 Stockholm, Sweden, 08-235480.

UNITED KINGDOM, Texas Instruments Ltd. Manton Lane, Bedford, England MK417PA, 0234-67466. A